ศรันย์ ไมตรีเวช

VBScript : ออบเจ็กต์โมเดลของไมโครซอฟท์

โครงสร้างออบเจ็กต์ที่เกี่ยวกับเวบของไมโครซอฟท์ เป็นสิ่งที่จะทำให้คุณเรียนรู้ VBScript ได้ด้วยความเข้าใจตลอดสาย

การศึกษาจุดใดจุดหนึ่งของภาษาโปรแกรมเพื่อนำไปใช้งานเฉพาะอย่างนั้น บางทีเป็นเรื่องสนุกน่าตื่นเต้น โดยเฉพาะเมื่อเราสามารถควบคุมเหตุการณ์ที่เกิดขึ้นในระดับเครือข่ายอินเทอร์เน็ต อย่างที่ผมสาธิตไปในตอนก่อน เกี่ยวกับเรื่องของการตรวจสอบอินพุตของผู้เข้าเยี่ยมว่ามีความสมบูรณ์ตรงใจเรามากน้อยเพียงใด

แต่เมื่อเกิดจุดประสงค์อื่นขึ้นมา ปัญหาที่ประสบก็คือ เมื่อเปิดตำราหรือหาแหล่งอ้างอิงแล้ว ไม่พบคำตอบที่จะนำมาสนองความต้องการ อย่างน้อยที่สุดก็ต้องสืบสาวค้นหาว่าจุดประสงค์ของเรามันน่าจะเข้าหมวดหมู่ใดในตำรา เมื่อพลิกดูอ่านแล้วก็เกิดความรู้สึกหลงทางยิ่งตะลุยอ่านตัวอย่างเพื่อจับทิศว่าควรไปทางไหน จึงจะโยงมาเกี่ยวข้องกับจุดประสงค์ของเรา ยิ่งรู้สึกมืดมน เพราะน้อยครั้งที่เราจะเผชิญพบ "ทางตรง" ที่เป็นคำตอบสำเร็จรูปตรงใจ

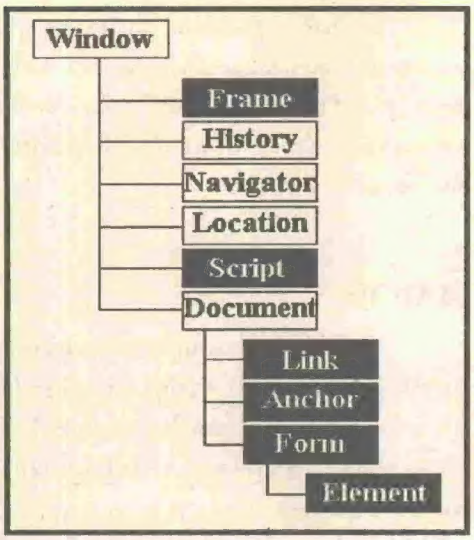
อย่างที่ผมตั้งสันนิษฐานไว้แต่ตอนแรกครับ ว่าคุณอ่านบทความ VBScript เพราะเคยเป็นโปรแกรมเมอร์วิชวลเบสิกมาก่อน ดังนั้นผมคิดว่าทางลัดที่สุดน่าจะได้แก่การตั้งความหวังว่าจะทำอะไรสักอย่างกับโฮมเพจของเรา โดยใช้ VBScript มาเป็นเครื่องมือ และเพื่อจับทิศได้ไว เราควรเริ่มด้วยการศึกษาออบเจ็กต์โมเดลของไมโครซอฟท์ก่อนเพื่อน เนื่องจากมันทำให้คุณมองเห็นภาพกว้างได้ถูกต้อง เมื่อตั้งเป้าประสงค์จะเขียนสั่งบราวเซอร์ให้ทำกิจใดสมควรคลำหาจากจุดไหน และจุดนั้นมีสายสัมพันธ์เชื่อมโยงกับจุดรอบนอกใดบ้าง

สิ่งที่น่าดีใจก็คือออบเจ็กต์โมเดลของไมโครซอฟท์เป็นไปตามครรลองเดียวกับของซันไมโครซิสเต็ม กล่าวคือ เมื่อคุณเข้าใจออบเจ็กต์โมเดลที่ผมจะกล่าวถึงนี้ ก็สามารถนำไปประยุกต์ใช้ได้ทั้งกับ VBScript และ JavaScript จะแตกต่างกันก็คือออบเจ็กต์โมเดลของไมโครซอฟท์ได้เพิ่มเอาสิ่งที่เรียกว่าสคริปต์ ActiveX (ต่อไปผมขอเรียกว่า "แอคทีฟสคริปต์") เข้ามาไว้ด้วย ซึ่งก็หมายความว่าการเรียนรู้ออบเจ็กต์โมเดลของไมโครซอฟท์จะครอบคลุมครบถ้วนกว่าการเรียนรู้ออบเจ็กต์โมเดลจากซันไมโครซิสเต็มซึ่งเป็นเจ้าตำรับเสียอีก

ภาพกว้างของออบเจ็กต์โมเดล

เมื่อพูดถึงออบเจ็กต์โมเดล ต่อไปนี้คุณต้องเห็นภาพเหมือนในใจเหมือนในรูปที่ 1 สำหรับออบเจ็กต์ที่ผมลงพื้นดำไว้เป็นออบเจ็กต์ที่มีได้หลายตัวภายใต้การครอบครองของออบเจ็กต์แม่ที่อยู่เหนือขึ้นไปหนึ่งลำดับ

1. Window เป็นออบเจ็กต์ใหญ่ที่สุดในบราวเซอร์ เพราะมันหมายถึงตัวบราวเซอร์เอง เปรียบกับวิชวลเบสิกแล้ว ก็เหมือนฟอร์ม App คือเป็นแอพพลิเคชันทั้งตัว เป็นหน้าต่างทั้งบานที่โปรแกรมเมอร์และผู้ใช้เห็นทุกสิ่งที่เราเห็นในบราวเซอร์ ล้วนแล้วแต่เป็นสมบัติของ Window ทั้งสิ้น



รูปที่ 1
ลำดับของออบเจ็กต์
ที่คุณควรนึกถึง
ให้ได้ขึ้นใจ

2. Frame เป็นออบเจ็กต์ที่มีได้หลายตัวในหนึ่งออบเจ็กต์วินโดว์ หนึ่งออบเจ็กต์เฟรมสามารถแสดงได้หนึ่งไฟล์ HTML เปรียบกับวิชวลเบสิกเดิมแล้ว เราคงต้องคิดถึงเรื่องของ MDI (Multiple Document Interface) ซึ่งแยกออกเป็น MDI Form และ MDI Child สำหรับ MDI Form นั้นเป็นหน้าต่างบานใหญ่สุดซึ่งรวมเอา MDI Child ไว้ในตัว ทำนองเดียวกับที่ Window รวมเอา Frame ไว้ภายในหลาย ๆ บานนั่นเอง ต่างแต่ว่าแต่ละเฟรมจะถูกแบ่งสัดส่วนด้วยเส้นกรอบซึ่งย่อขยายได้ ส่วนฟอร์ม MDI ตัวลูกจะเห็นเป็นหน้าต่างอิสระ อันนี้เป็นการเปรียบเทียบให้เห็นพอคร่าว ๆ นะครับ การใช้จริงจะแตกต่างกันมาก

3. History เป็นออบเจ็กต์ที่มีได้เพียงหนึ่งเดียว โดยออบเจ็กต์นี้จะมีลิสต์ของเวบเพจที่เคยถูกแสดงในออบเจ็กต์วินโดว์ ซึ่งแน่นอนมีจำนวนจำกัดแบบเก่าไปใหม่มาแทนที่กัน สำหรับผู้ใช้ทั่วไปต้องเรียกหน้าต่าง History ขึ้นมาเพื่อย้อนกลับไปดูเวบเพจซึ่งเคยบราวซ์ไว้ แต่สำหรับโปรแกรมเมอร์ เราอาจใช้ประโยชน์จากออบเจ็กต์นี้ได้หลากหลาย เป็นต้นว่าพาผู้ใช้กลับไปยังเวบเพจที่เราเช็คแล้วว่าตรงกับจุดประสงค์บางอย่างได้โดยอัตโนมัติ

4. Navigator เรามีออบเจ็กต์นี้ไว้เรียกดูข้อมูลเกี่ยวกับตัวบราวเซอร์โดยเฉพาะ เป็นข้อมูลของบราวเซอร์นะครับไม่ใช่ของเวบเพจ ยกตัวอย่างเช่น เราอยากรู้ว่าบราวเซอร์ที่ผู้ใช้มีอยู่เป็นเวอร์ชันอะไร เพื่อตัดสินใจได้ถูกว่าควรหรือไม่ควรปฏิบัติการใดให้เหมาะกับเวอร์ชัน เพราะเวอร์ชันแรก ๆ ของบราวเซอร์ทุกค่ายมักมีข้อจำกัด โดยเฉพาะอย่างยิ่งไม่สามารถปฏิบัติตามกฎของ HTML เวอร์ชันใหม่ ๆ (ล่าสุดคือ 3.2)

5. Location ออบเจ็กต์นี้แสดงให้เราเห็นว่าบราวเซอร์รู้จักกับเวบไซท์ที่มันกำลังแสดงมากน้อยเพียงใด ตัวอย่างเช่น มันเชื่อมต่อเข้าทางพอร์ตไหน ชื่อโฮสต์ของเวบไซท์เป็นอะไร ใช้โปรโตคอลใด (เช่น http: หรือ ftp:)

6. Script อย่าเพิ่งสับสนนะครับ ระหว่างออบเจ็กต์สคริปต์กับตัวสคริปต์ที่เราเขียนขึ้นมา ตามความรู้สึกธรรมดาของเรา เมื่อเขียน VBScript ปะปนรวมกันกับสคริปต์ HTML ก็น่าจะหมายถึงมันอยู่ในออบเจ็กต์เช่น document แต่ความจริงแล้วออบเจ็กต์สคริปต์อาศัยออบเจ็กต์วินโดว์เป็นที่เกิดครับ คิดเทียบกับวิชวลเบสิกเดิมแล้ว ออบเจ็กต์สคริปต์ก็คือหนึ่งซื่อโพรซีเดอร์หรือฟังก์ชันนั่นเอง ต่อไปคุณจะได้เห็นว่าเราเรียกสคริปต์หนึ่ง ๆ ขึ้นมาผ่านชื่อซึ่งตั้งขึ้นใช้ในฐานะออบเจ็กต์สคริปต์ ให้ความรู้สึกไม่ต่างกับการเรียกโพรซีเดอร์ขึ้นมานั่นเอง

7. Document เป็นออบเจ็กต์ที่เก็บรายละเอียดเกี่ยวกับเวบเพจหรือไฟล์ HTML ที่มันกำลังแสดง ไม่ว่าจะเป็นข้อความธรรมดา ลิงค์ ฟอร์ม ฯลฯ ทั้งหมดที่อยู่ในหน้าเวบเพจนั้นนั่นเอง ต่อไปคุณจะเห็นว่าอะไรเช่นฟอร์มนั้น ก็สามารถเป็นออบเจ็กต์อะเรย์ คุณระบุใช้-ปรับเปลี่ยนได้ผ่านเลขอินเด็กซ์ คล้ายกับที่ทำในวิชวลเบสิกเดิมตอนจัดการกับคอนโทรลอะเรย์น่ะครับ

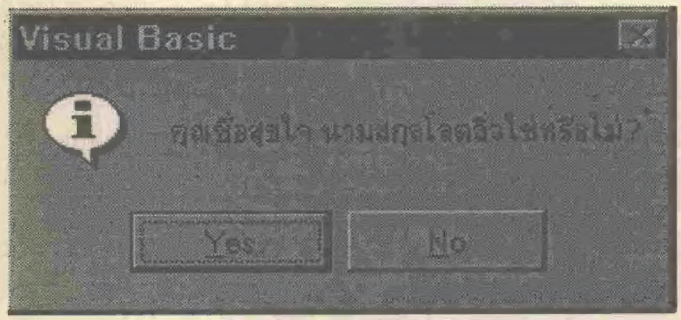
ออบเจ็กต์ Document มีเรื่องให้น่าพูดถึงเยอะ เพราะค่อนข้างมีโครงสร้างภายในซับซ้อน เนื่องจากมันมีออบเจ็กต์ลูกอีกหลายตัวคือ

- Link เป็นออบเจ็กต์ที่ให้คุณอ่านได้ทางเดียว คือเราได้ output ด้วยการอ่านค่าอย่างเดียว แต่ปรับเปลี่ยนหรือกำหนดให้มันไม่ได้ เป็นออบเจ็กต์อะเรย์ได้เหมือนฟอร์ม คุณจะเห็นความสำคัญของมันตอนเกิดอีเวนต์อะไรบางอย่าง เช่น เมื่อผู้ใช้นำพอยน์เตอร์ไปวางลอยเหนือลิงค์ คุณอาจแสดงข้อความพิเศษเพื่อดึงดูดใจให้รีบคลิ๊ก หรือบอกกล่าวสะกิดเตือนว่าลิงค์นี้ควรไม่ควรสำหรับบุคคล-กาลเทศะแบบใดเป็นต้น

- Anchor เป็นออบเจ็กต์อะเรย์ที่มีค่าอ่านได้ทางเดียวเหมือน Link เอาไว้ดูน่ะครับว่าชื่อของ "ปลายทาง" ที่ตั้งไว้เป็นอะไร เช่น ในเวบเพจเดียวกัน คุณอาจลง Anchor ไว้ 2 จุดด้วยแอตทริบิวต์ Name ของแท็ก <A> เราก็จะมีออบเจ็กต์ Anchor เป็น document.anchors(0) และ document.anchors(1) อย่างนี้เป็นต้น คุณมีรายละเอียดเกี่ยวกับออบเจ็กต์ Anchor เพียงหนึ่งเดียวคือชื่อของมันครับ

- Form ต้องจำไว้ไม่ให้สับสนกับเรื่องสคริปต์นะครับ ออบเจ็กต์ฟอร์มเป็นลูกของออบเจ็กต์ document เกิดขึ้นบน document คุณจะสามารถควบคุมอินพุตที่ได้จากผู้เข้าเยี่ยมเวบในหลายต่อหลายทางผ่านออบเจ็กต์ฟอร์มนี้ แต่ก่อนอื่นคุณต้องมีพื้นฐานเกี่ยวกับฟอร์มเป็นอย่างดี ซึ่งเมื่อคราวก่อนผมก็พูดหลัก ๆ ไปแล้ว แต่อาจไม่ครอบคลุมถ้วนทั่วนัก

ในฟอร์มยังมีออบเจ็กต์ลูกตัวหนึ่งชื่อ Element ออบเจ็กต์นี้แหละครับที่เปรียบเหมือนคอนโทรลในวิชวลเบสิกเดิม กล่าวคือ สิ่งที่ปรากฏในฟอร์ม เช่น ปุ่ม กล่องรับข้อความ กรอบรับเรื่อง กล่องตัวเลือก ฯลฯ ล้วนแล้วแต่มีศักดิ์เป็น Element ทั้งสิ้น แน่นอนเมื่อมีได้หลายตัวย่อมเป็นออบเจ็กต์อะเรย์ เวลาใช้จริงจะรู้สึกว่าซับซ้อน เหตุเพราะต้องเขียนเจาะลำดับชั้นออบเจ็กต์กันมาเยอะกว่าจะถึงตัว ถึงพรอพเพอร์ตี้ของมัน



รูปที่ 2 เมสเสจบ็อกซ์ขอการยืนยันเกี่ยวกับอินพุตที่ป้อนเข้ามาก่อนส่งจริง

เราได้อะไรจากการศึกษาออบเจ็กต์โมเดล?

ถึงตอนนี้ เมื่อเขียนโค้ด หรือนึกถึงการปรับเปลี่ยนหน้าตา หรือเรียกดูค่าหนึ่งค่าใดในบราวเซอร์ คุณจะมองออกแล้วไงครับว่าควรเจาะลำดับออบเจ็กต์ลงไปท่าไหน ยกตัวอย่างเช่นเมื่อคุณเกิดคำถามขึ้นในใจว่า "เราจะถามขอคำยืนยันจากผู้เข้าเยี่ยมได้อย่างไรว่าอินพุตที่เขากรอกลงไปมันตรงความเป็นจริง?" คุณอาจนึกลำดับเหตุการณ์เช่นเมื่อผู้เข้าเยี่ยมเขาคลิ๊กปุ่ม Submit แล้วให้มีเมสเสจบ็อกซ์ถามหาความแน่ใจเช่นรูปที่ 2

ซึ่งเป้าหมายนี้ แน่นอนคุณต้องนึกถึงข้อความหรือสตริงตายตัว เช่น "คุณชื่อ", "นามสกุล" และ "ใช่หรือไม่?" ส่วนค่าสตริงที่แปรไปตามอินพุตของผู้เข้าเยี่ยมก็ได้แก่ "สุขใจ" และ "โสดลิ่ว" ซึ่งสองตัวนี้แหละคุณต้องการมันจากกล่องรับข้อความ ซึ่งอาจแยกเป็นกล่องรับชื่อและกล่องรับนามสกุลโดยเฉพาะ เขียนได้คือ

```
<FORM NAME="VisitorInfo">
<INPUT NAME="realname" TYPE=TEXT SIZE=40>
<INPUT NAME="surname" TYPE=TEXT SIZE=40>
<INPUT NAME="ToSubmit" TYPE=Button>
</FORM>
```

เมื่อเข้าโพรซีเดอร์ที่ถูกเรียกเนื่องจากผู้ใช้คลิ๊กปุ่ม Submit คุณก็อาจเขียนโค้ดดักไว้ว่า

```
<SCRIPT FOR="ToSubmit" EVENT="onClick">
Response=MsgBox "คุณชื่อ" &
document.VisitorInfo.realname.value & "นามสกุล" &
document.VisitorInfo.surname.value & "ใช่หรือไม่?",
4+64

if Response=6 then VisitorInfo.Submit
</SCRIPT>
```

คุณใช้ความรู้เดิมเกี่ยวกับ MsgBox ในวิชวลเบสิก ผสมกับการใช้เมธอด Submit ซึ่งผมกล่าวถึงไปในตอนที่แล้ว ก็จะเขียนโค้ดออกมาได้ตามที่เห็น

คำถามคือ document.VisitorInfo.realname.value และ document.VisitorInfo.surname.value นั้น เราได้มาอย่างไร ก็จากการทราบโครงสร้างออบเจ็กต์โมเดลไงครับว่าฟอร์มอยู่ได้ document และ Element อยู่ได้ฟอร์มอีกที การเข้าถึงข้อมูลตามลำดับก็เขียนไปเหมือนอย่างที่เคยรู้ ๆ กันมานั่นเอง

ทิ้งท้าย

จะเห็นว่าการศึกษาแก่นสำคัญของแต่ละภาษาโปรแกรมเชิงวัตถุให้เข้าใจ จะทำให้เราจับทิศทางการเขียนโค้ดได้ไวกว่ามานั่งเปิดดูทีละจุดที่สนใจ ในคราวต่อไปผมจะแสดงให้เห็นว่าเราใช้ความรู้ในเชิงออบเจ็กต์ปรับปรุงโฮมเพจให้ปฏิบัติการดังใจได้มากกว่านี้เยอะแค่ไหนครับ สวัสดี